

По-сложни проверки

Вложени if конструкции и
по-сложни логически условия

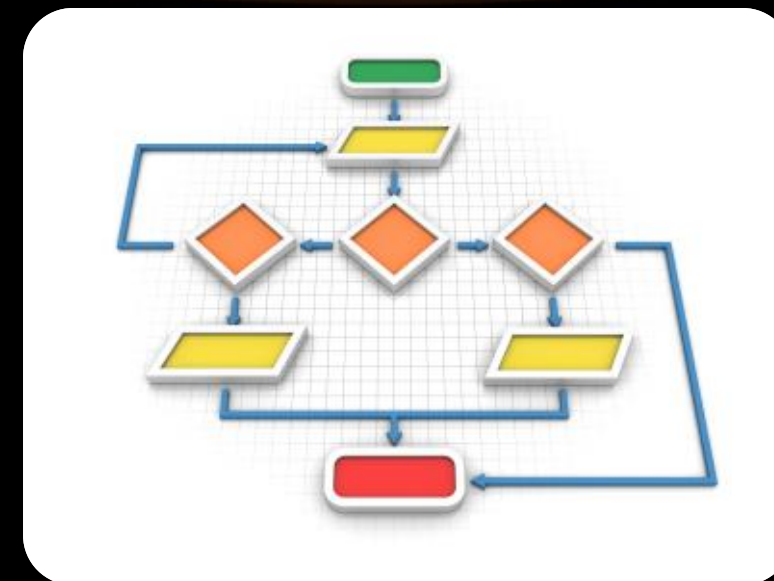


СофтУни

трейнерски екип

Софтуерен университет

<http://softuni.bg>



Съдържание

1. Вложени проверки
2. По-сложни проверки
 - Логическо "и", "или", отрицание и приоритет на условия
3. Switch-case конструкция





Вложени проверки

If-конструкции, вложени една в друга

Вложени проверки

- Конструкциите **if-else** могат да се влагат една в друга:

```
if (condition1) {  
  console.log("condition1 valid");  
  if (condition2)  
    console.log("condition2 valid");  
  else  
    console.log("condition2 not valid");  
}
```

Вложена **if**
конструкция

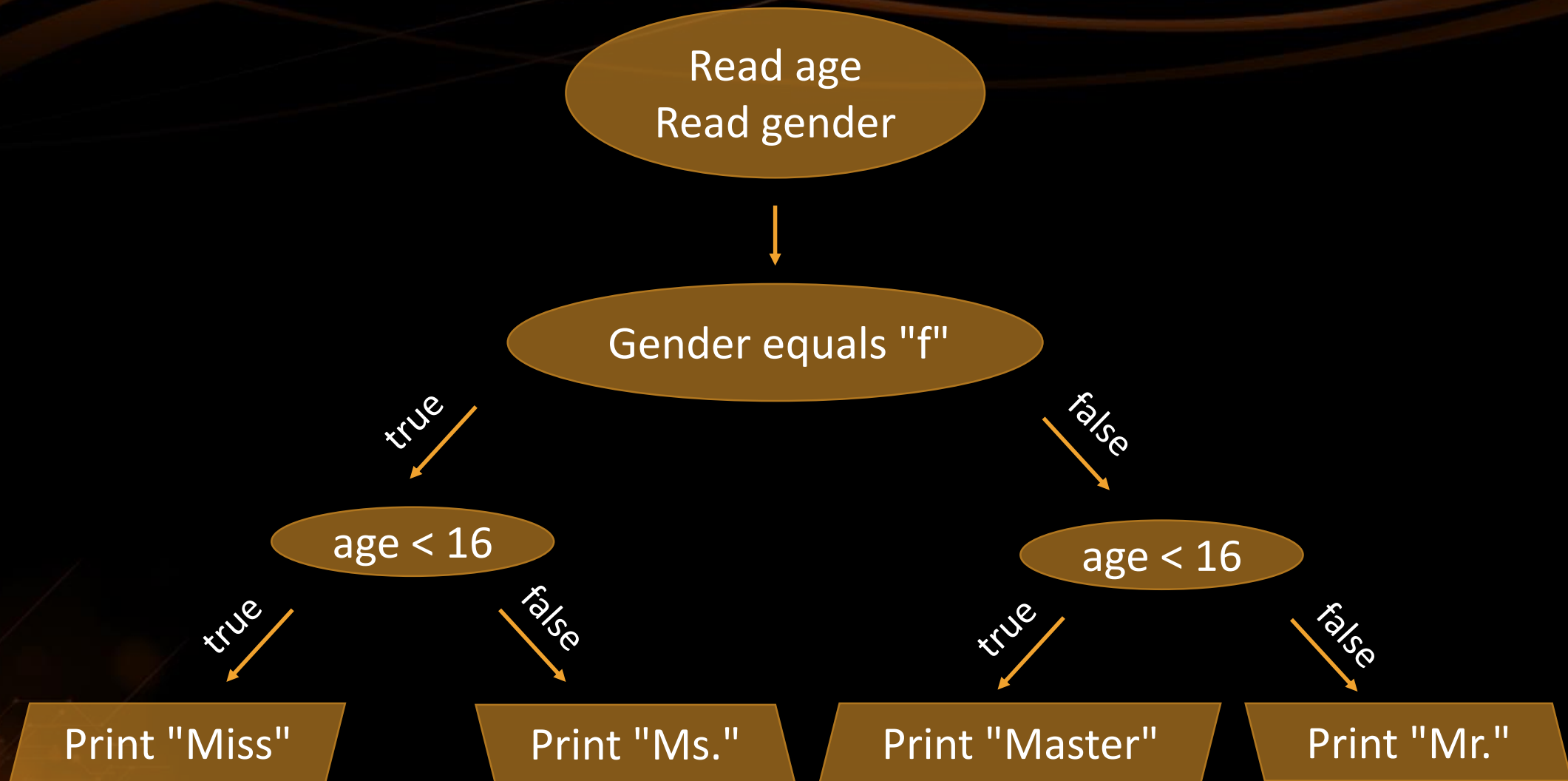
- Само при изпълнение на първото условие се преминава към вложената проверка

Обръщение според възраст и пол – условие

- Напишете програма, която:
 - Чете от потребителя:
 - Възраст
 - Пол
 - Принтира обръщение според въведеното, както е показано на **схемата** (в следващия слайд)
- Примерен вход и изход:



Обръщение според възраст и пол – условие (2)



Обръщение според възраст и пол - решение

```
function personalTitles([arg1, arg2]) {  
  let age = Number(arg1);  
  let gender = arg2;  
  if (gender == "f") {  
    if (age < 16) console.log("Miss");  
    else console.log("Ms.");  
  } else if (gender == "m") {  
    if (age < 16) console.log("Master");  
    else console.log("Mr.");  
  }  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#0>

Квартално магазинче – условие

- Напишете програма, която:
 - Чете от потребителя:
 - Име на продукт
 - Град
 - Количество на продукта
 - Пресмята цената му спрямо таблицата:

| град / продукт | coffee | water | beer | sweets | peanuts |
|----------------|--------|-------|------|--------|---------|
| Sofia | 0.50 | 0.80 | 1.20 | 1.45 | 1.60 |
| Plovdiv | 0.40 | 0.70 | 1.15 | 1.30 | 1.50 |
| Varna | 0.45 | 0.70 | 1.10 | 1.35 | 1.55 |

Квартално магазинче – условие (2)

- Примерен вход и изход:



Квартално магазинче - решение

```
function smallShop([arg1, arg2, arg3]) {  
  let product = arg1.toLowerCase();  
  let town = arg2.toLowerCase();  
  let quantity = Number(arg3);  
  if (town == "sofia") {  
    if (product == "coffee")  
      console.log(0.50 * quantity);  
    // TODO: Check the other cases ...  
  }  
  if (town == "varna") // TODO: finish this ...  
  if (town == "plovdiv") // TODO: finish this ...  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#1>



&&, ||, !

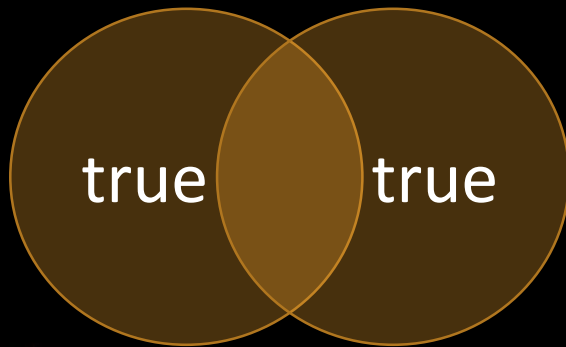
По-сложни проверки

Булеви оператори

Булеви оператори

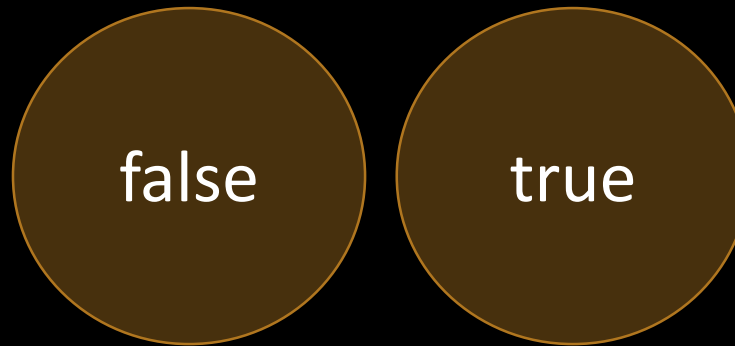
- Оператори, които комбинират или изключват условия
- Връщат булеви резултат (**true** или **false**)

"&&" - И



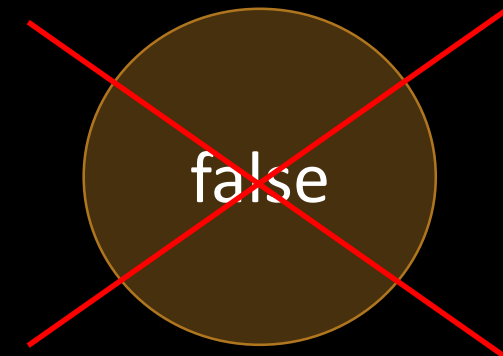
Вярност на двете условия

"||" - ИЛИ



Вярност на
едното **или** другото
условие

"!" - ОТРИЦАНИЕ



Отрицание на условие

Логическо "И"

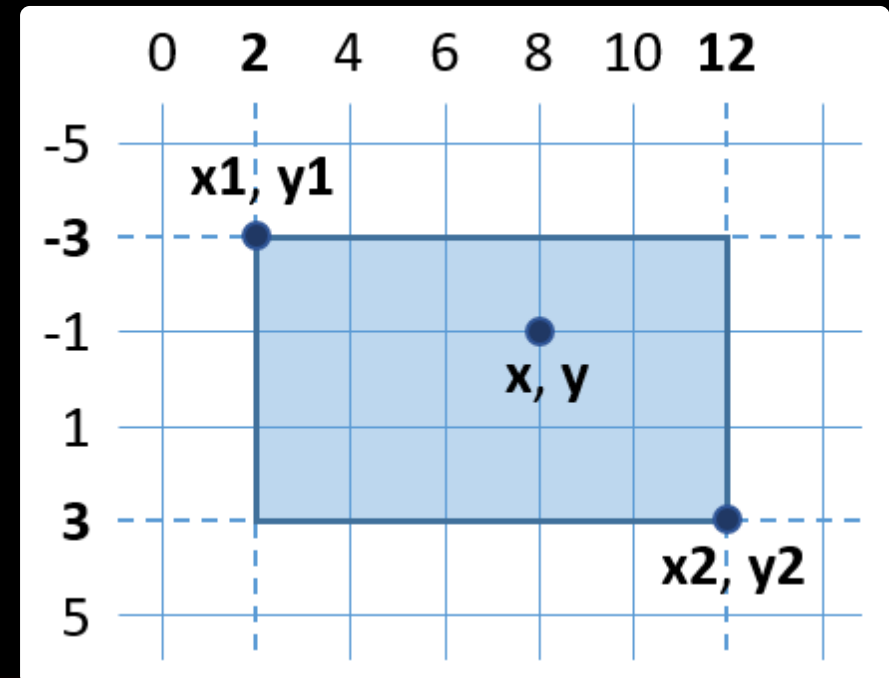
- Проверява изпълнението на няколко условия **едновременно**
- Пример: проверка дали число е едновременно:
 - по-голямо от 5 и по-малко от 10
 - четно

&&

```
let a = parseInt(arg1);  
if (a > 5 && x < 10 && a % 2 == 0) ...
```

Точка в правоъгълник - условие

- Напишете програма, която:
 - Проверява дали точка е **вътрешна** за даден правоъгълник
- Точка е вътрешна, ако е **едновременно**:
 - надясно от лявата му страна
 - наляво от дясната му страна
 - надолу от горната му страна
 - нагоре от долната му страна



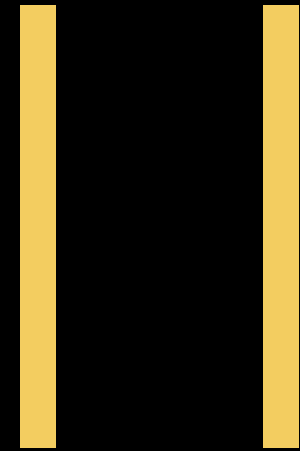
Точка в правоъгълник - решение

```
function pointInRectangle([arg1, arg2, arg3, arg4, arg5, arg6]) {  
  // TODO: Read the coordinates of the points ...  
  let x = Number(arg5), y = Number(arg6);  
  if (x >= x1 && x <= x2 && y >= y1 && y <= y2) {  
    console.log("Inside");  
  }  
  else {  
    console.log("Outside");  
  }  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#2>

Логическо "ИЛИ"

- Проверява дали е изпълнено поне едно измежду няколко условия
- Пример: проверка дали въведената дума е:
 - "Example" или "Demo"



```
let word = arg1;  
if (word == "Example" || word == "Demo") ...
```


Плод или зеленчук - условие

- Напишете програма, която:
 - Проверява дали въведенният вход от потребителя е плод или зеленчук измежду изброените:
 - Плодове: banana, apple, kiwi, cherry, lemon, grapes
 - Зеленчуци: tomato, cucumber, pepper, carrot
 - Извежда "vegetable", "fruit" или "unknown"
- Примерен вход и изход:

lemon



fruit

carrot



vegetable

java



unknown

Плод или зеленчук - решение

```
function fruitOrVegetable([arg1]) {  
  let p = arg1;  
  if (p == "banana" || p == "apple" || p == "kiwi" ||  
      p == "cherry" || p == "lemon" || p == "grapes")  
    console.log("fruit");  
  else if (p == "tomato" || p == "cucumber" ||  
          p == "pepper" || p == "carrot")  
    console.log("vegetable");  
  else  
    console.log("unknown");  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#3>

Логическо отрицание

- Проверява дали **не е изпълнено** дадено условие
- Пример:
 - Проверка дали число е по-голямо от 10 и е четно:

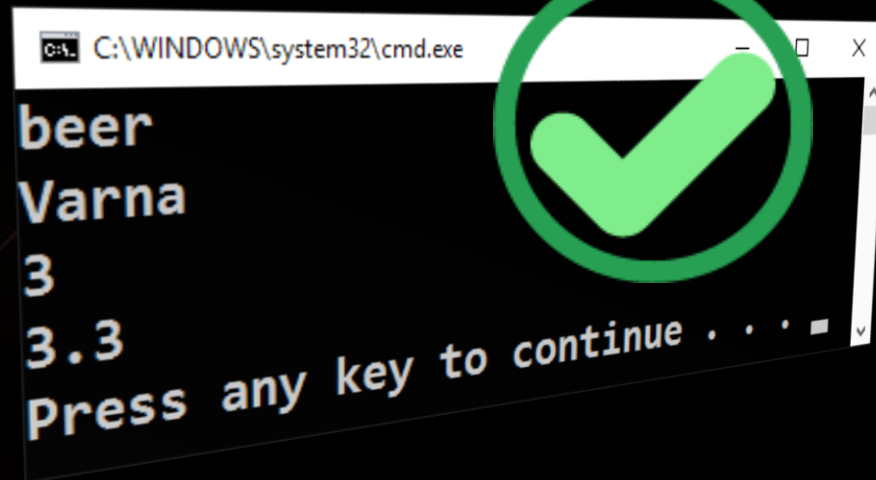


```
let isValid = (number > 10) && (number % 2 == 0);  
if (!isValid) {  
    console.log("Invalid");  
}
```

Приоритет на условията

- Чрез скоби () можем да приоритизираме условия
- Пример:
 - Проверка дали да се интервюира кандидат за позиция

```
let collegeYears = 4;  
let averageGrade = 4.75;  
let yearsOfExperience = 2;  
if (collegeYears > 3 && (averageGrade >= 4 || yearsOfExperience > 1)) {  
    console.log("Interview applicant");  
}
```



```
C:\WINDOWS\system32\cmd.exe  
beer  
Varna  
3  
3.3  
Press any key to continue . . .
```

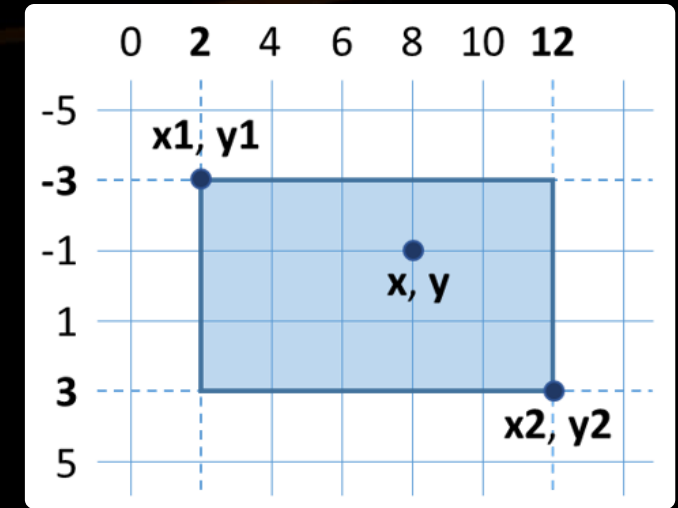


Логически оператори

Работа на живо в клас (лаб)

Точка върху страна на правоъгълник - условие

- Напишете програма, която:
 - Чете 6 десетични числа x_1 , y_1 , x_2 , y_2 , x и y
 - Извежда дали точката е:
 - върху страна от правоъгълника ("Border")
 - в или извън правоъгълника ("Inside/Outside")



- Примерен вход и изход:

```
2
-3
12
3
8
-1
```



```
Inside /
Outside
```

```
2
-3
12
3
12
-1
```

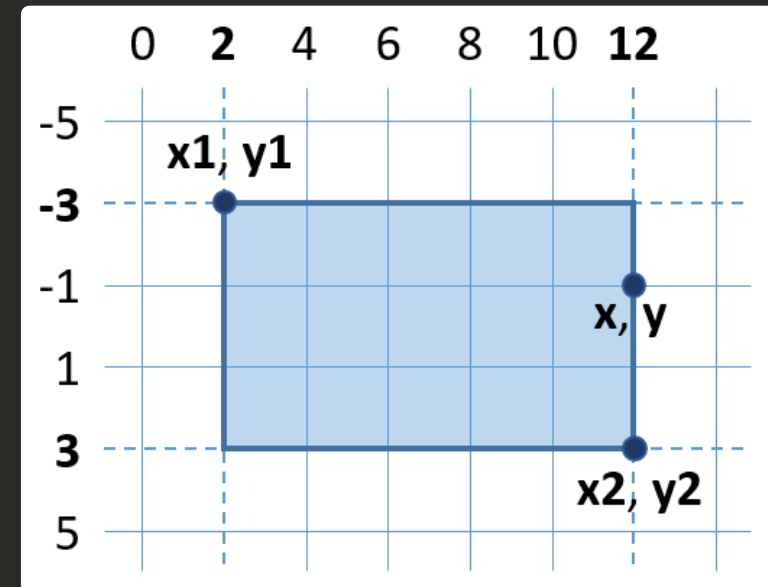


```
Border
```

По-сложни логически условия

- Точка лежи върху някоя от страните на правоъгълник, ако:
 - x съвпада с $x1$ или $x2$ и същевременно y е между $y1$ и $y2$ или
 - y съвпада с $y1$ или $y2$ и същевременно x е между $x1$ и $x2$

```
if (((x == x1 || x == x2) &&  
    (y >= y1) && (y <= y2)) ||  
    ((y == y1 || y == y2) &&  
    (x >= x1) && (x <= x2))) {  
    console.log("Border");  
}
```



Опростиране на логически условия

- Предходното условие може да се опрости ето така:

```
function isOnBorder([arg1, arg2, arg3, arg4, arg5, arg6]) {  
  // TODO: Read the input points  
  let onLeftSide = (x == x1) && (y >= y1) && (y <= y2);  
  let onRightSide = (x == x2) && (y >= y1) && (y <= y2);  
  let onUpSide = (y == y1) && (x >= x1) && (x <= x2);  
  let onDownSide = (y == y2) && (x >= x1) && (x <= x2);  
  
  if (onLeftSide || onRightSide || onUpSide || onDownSide)  
    console.log("Border");  
  else  
    console.log("Inside / Outside");  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#5>

Магазин за плодове – условие

- Напишете програма, която:
 - Чете потребителски вход:
 - Продукт
 - Ден
 - Количество
 - Извежда сумата, която трябва да се заплати **според деня и продукта**

Магазин за плодове – условие (2)

- В работни дни продава на следните цени:

| плод | banana | apple | orange | grapefruit | kiwi | pineapple | grapes |
|------|--------|-------|--------|------------|------|-----------|--------|
| цена | 2.50 | 1.20 | 0.85 | 1.45 | 2.70 | 5.50 | 3.85 |

- В почивни дни цените са по-високи:

| плод | banana | apple | orange | grapefruit | kiwi | pineapple | grapes |
|------|--------|-------|--------|------------|------|-----------|--------|
| цена | 2.70 | 1.25 | 0.90 | 1.60 | 3.00 | 5.60 | 4.20 |

- Примерен вход и изход:



Магазин за плодове - решение

```
function fruitShop([arg1, arg2, arg3]) {  
  let fruit = arg1;  
  let day = arg2;  
  let quantity = Number(arg3);  
  if (day == "saturday" || day == "sunday") {  
    if (fruit == "banana") price = 2.70;  
    else if (fruit == "apple") price = 1.25;  
    // TODO: more fruits come here ...  
  }  
  else if (day == "monday" || day == "tuesday" || day ==  
    "wednesday" || day == "thursday" || day == "friday") {  
    if (fruit == "banana") price = 2.50;  
    // TODO: more fruits come here ...  
  }  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#6>

Търговски комисионни - условие

- Напишете програма, която:
 - Чете от потребителя:
 - Град
 - Обем на продажби (десетично число)
 - Изчислява **комисионната**, която дадена фирма дава на търговците според града и обема на продажбите
 - Извежда **стойността** на комисионната, закръглена до 2 цифри след десетичната запетая

ТЪРГОВСКИ КОМИСИОННИ – УСЛОВИЕ (2)

| Град / цена | $0 \leq s \leq 500$ | $500 < s \leq 1\,000$ | $1\,000 < s \leq 10\,000$ | $s > 10\,000$ |
|-------------|---------------------|-----------------------|---------------------------|---------------|
| Sofia | 5% | 7% | 8% | 12% |
| Varna | 4.5% | 7.5% | 10% | 13% |
| Plovdiv | 5.5% | 8% | 12% | 14.5% |

- Примерен вход и изход:



ТЪРГОВСКИ КОМИСИОННИ - решение

```
function tradeComissions([arg1, arg2]) {
  let town = arg1.toLowerCase();
  let sales = Number(arg2);
  let comission = -1.0;
  if (town == "sofia") {
    if (0 <= sales && sales <= 500) comission = 0.05;
    else if (500 < sales && sales <= 1000) comission = 0.07;
    // TODO: check the other price ranges ...
  }
  else if (town == "varna") // TODO: check the price ranges ...
  else if (town == "plovdiv") // TODO: check the price ranges ...

  if (comission >= 0)
    console.log((sales * comission).toFixed(2));
  else console.log("error");
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#7>



switch (...)
case ...?

Условна конструкция Switch-case

По-доброто If-Else-If-Else?

Условна конструкция Switch-case

- Работи като поредица `if/else-if/else-if...`

Условието в `switch-case` е стойност

```
switch (...){
```

```
case ...:  
//code  
break;
```

```
case ...:  
//code  
break;
```

```
default:  
//code  
break;
```

```
}
```

Изброяване на условия (стойности) за проверката

Обхваща всички случаи, които не са описани по-горе

Ден от седмицата - условие

- Напишете програма, която:
 - Чете **цяло число** въведено от потребителя
 - Извежда **деня от седмицата с текст** (на английски) според въведеното **число (1...7)**
 - Извежда "Error!", ако числото не е в диапазона
- Примерен вход и изход:



Ден от седмицата - решение

```
function dayOfWeek([arg1]) {  
  let day = parseInt(arg1);  
  switch (day) {  
    case 1: console.log("Monday"); break;  
    case 2: console.log("Tuesday"); break;  
    //TODO: more cases come here  
    case 7: console.log("Sunday"); break;  
    default: console.log("Error!"); break;  
  }  
}
```

Множество случаи в Switch-case

- Чрез switch-case, можем да изпълняваме един и същ код за множество условия

```
switch (...) {  
    case ... :  
    case ... :  
    case ... :  
        //code  
        break;  
    default:  
        //code  
        break;  
}
```

Кодът ще се изпълни за
някое от трите условия в
серията

Вид животно - условие

- Напишете програма, която:
 - Чете от потребителя дума (животно)
 - Възможен вход: "dog", "crocodile", "tortoise", "snake":
 - Извежда **вида на животно според името му**
 - Бозайник – "mammal"
 - Влечуго – "reptile"
 - Други – "unknown"
- Примерен вход и изход:

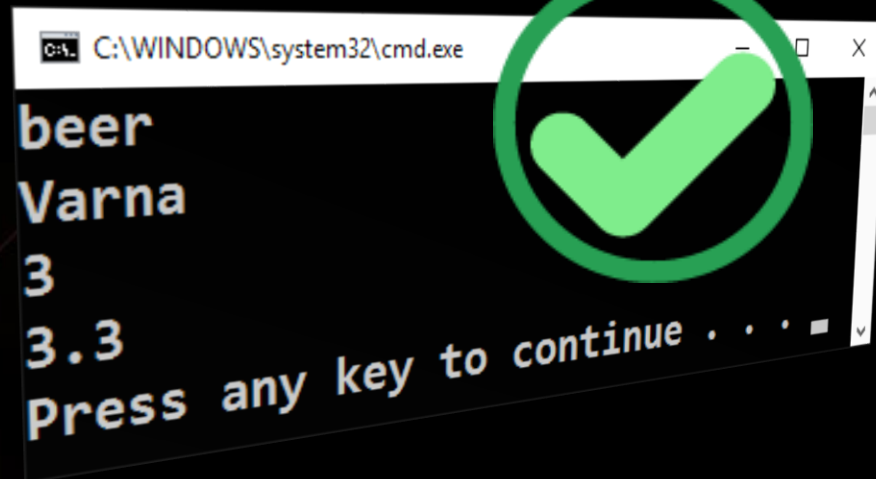
dog → mammal

car → unknown

Вид животно - решение

```
function animalType([arg1]) {  
  let animal = arg1;  
  switch (animal) {  
    case "dog": console.log("mammal"); break;  
    case "crocodile": //TODO: print result  
    case "tortoise": //TODO: print result  
    case "snake": console.log("reptile"); break;  
    default: console.log("unknown"); break;  
  }  
}
```

Тестване на решението: <https://judge.softuni.bg/Contests/Practice/Index/153#9>



```
C:\WINDOWS\system32\cmd.exe  
beer  
Varna  
3  
3.3  
Press any key to continue . . .
```



По-сложни проверки

Работа на живо в клас (лаб)



Точка и правоъгълник

Графично (GUI) приложение

How Blank Node.js Console Application project

The screenshot shows the 'New Project' dialog in Visual Studio. The 'JavaScript' category is expanded, and 'Blank Node.js Console Application' is selected. The project name is 'point-in-rectangle' and the location is 'C:\Projects\DesktopApps\'. A preview window shows the code 'console.log('Hello world');'.

New Project

Recent

Installed

Templates

- Visual C#
- Visual Basic
- JavaScript

 - Apache Cordova Apps
 - Node.js
 - Windows

- Visual F#
- Visual C++
- SQL Server
- Python
- TypeScript
- Game
- Build Accelerator
- Other Project Types

Samples

Online [Click here to go online and find templates.](#)

Name:

Location:

Solution name:

Create directory for solution

Create new Git repository

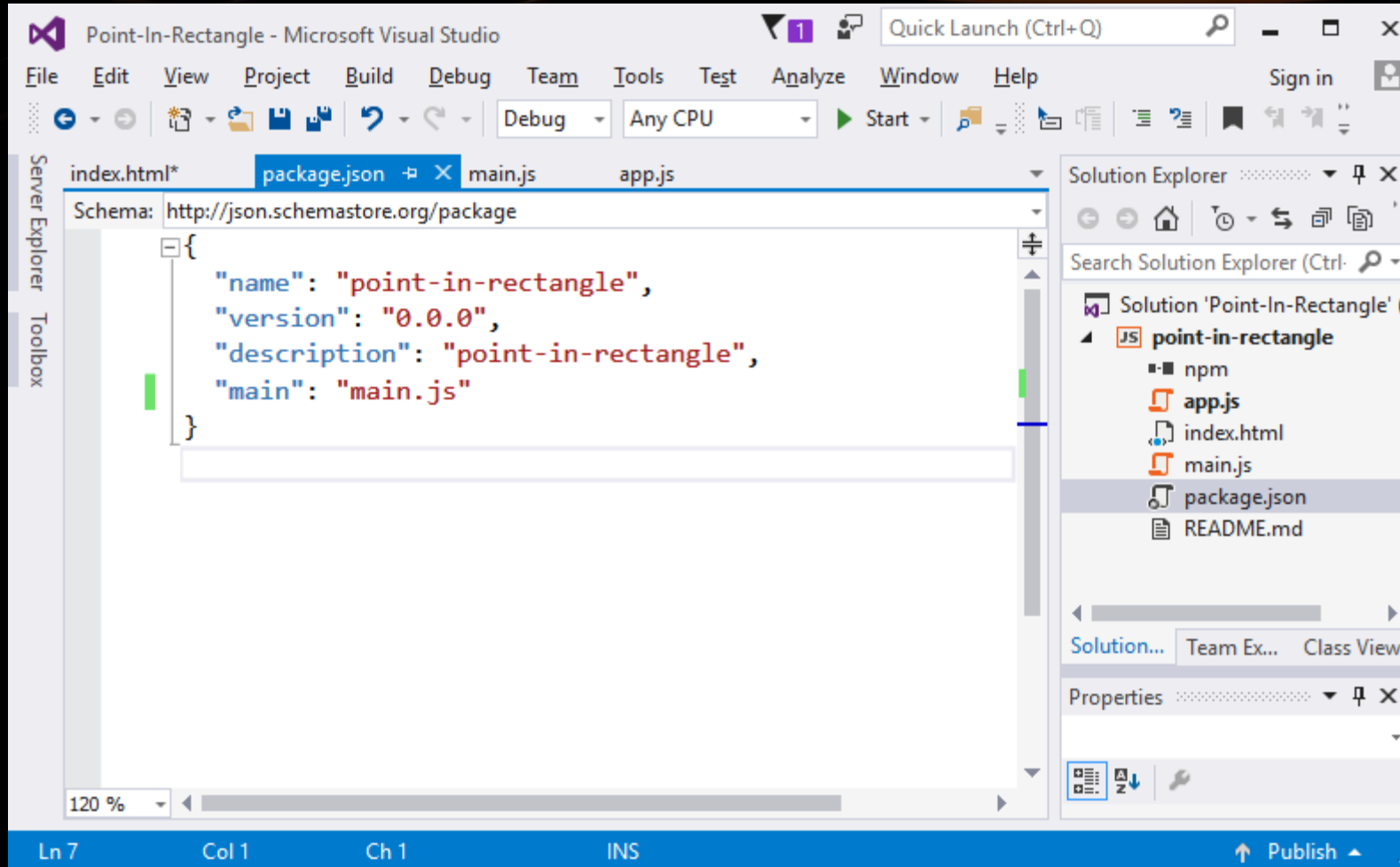
Search Installed Templates (🔍)

Type: JavaScript

An empty Node.js application.

```
app.js
console.log('Hello world');
```


Подготвяне на структурата на приложението



The screenshot shows the Visual Studio IDE with the following details:

- Title Bar:** Point-In-Rectangle - Microsoft Visual Studio
- Menu Bar:** File, Edit, View, Project, Build, Debug, Team, Tools, Test, Analyze, Window, Help
- Toolbar:** Includes icons for file operations, a 'Debug' dropdown, 'Any CPU' target, and a 'Start' button.
- Editor:** Displays the content of `package.json` with the following JSON structure:

```
{
  "name": "point-in-rectangle",
  "version": "0.0.0",
  "description": "point-in-rectangle",
  "main": "main.js"
}
```
- Solution Explorer:** Shows the project structure for 'Solution 'Point-In-Rectangle'':
 - point-in-rectangle
 - npm
 - app.js
 - index.html
 - main.js
 - package.json (selected)
 - README.md
- Status Bar:** Shows 'Ln 7 Col 1 Ch 1 INS' and a 'Publish' button.

Основни елементи в index.html

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta charset="utf-8" />
  <title>Point in Rectangle</title>
  <script src="app.js" type="text/javascript"></script>
</head>
```

Подаване на път към `app.js`, който съдържа функцията `draw()`

Закачане на функцията `draw()` към `onclick` събитието на бутона [Draw!]

```
<input type="button" onclick="draw()" value="Draw!" />
```

```
<div id="result">
  <label>Status:</label>
  <span id="status"></span>
</div>
```

Създаване на елементи, които да пазят резултата от функцията

Създаване на `canvas` елемент за чертаене на фигури

```
<div style="float:right">
  <canvas style="border: 2px solid orange;" id="a" width="400" height="200">
    This text is displayed if your browser does not support HTML5 Canvas.
  </canvas>
</div>
```

Взимане на координати и рисуване на екрана

```
//Get input for rectangle coordinates
let rectX1 = Number(document.getElementById("rect-x1").value) * 10;
let rectY1 = Number(document.getElementById("rect-y1").value) * 10;
// TODO: Get the coordinates of the second point
```

Вземете координатите на
точките на правоъгълника

```
//TODO: Calculate rectWidth and rectHeight

//Set rectangle style
context.strokeStyle = "#ff0000";
context.lineWidth = 3;

//Draw rectangle with given parameters
context.strokeRect(rectX1, rectY1, rectWidth, rectHeight);
```

Пресметнете големината на
страните на правоъгълника

Задайте стил на
правоъгълника и го
начертайте със `.strokeRect()`

Извършване на проверките

```
//Assign variables to (<div id="result">) and (<span id="status">)
html elements
```

```
let result = document.getElementById("status");
let output = document.getElementById("result");
```

Запазете в променливи
елементите от html кода,
които съдържат резултата от
проверката

```
//Check point position
```

```
if () {
    result.innerHTML = "Inside";
    output.style.backgroundColor = "palegreen";
}
else if () {
    result.innerHTML = "Border";
    output.style.backgroundColor = "gold";
}
else {
    result.innerHTML = "Outside";
    output.style.backgroundColor = "lightsalmon";
}
```

Допишете условията в
if-проверките



Точка и правоъгълник

Работа на живо в клас (лаб)

Какво научихме днес?

- Вложени проверки:

```
if (condition1) {  
    if (condition2) ...  
    else ...  
}
```



- По-сложни проверки с **&&**, **||**, **!** и **()**

```
if ((x == left || x == right) && y >= top && y <= bottom)  
    console.log("Point on the left or right side.");
```

Какво научихме днес? (2)

- Switch-case конструкция

```
switch (...){  
  case ...:  
    //code  
    break;  
  case ...:  
    //code  
    break;  
  default:  
    //code  
    break;  
}
```



По-сложни проверки



Въпроси?



- Настоящият курс (слайдове, примери, видео, задачи и др.) се разпространяват под свободен лиценз "Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International"



Безплатни обучения в СофтУни

- Фондация "Софтуерен университет" – softuni.org
- Софтуерен университет – качествено образование, професия и работа за софтуерни инженери
 - softuni.bg
- СофтУни @ Facebook
 - facebook.com/SoftwareUniversity
- СофтУни форуми – forum.softuni.bg

